

Aplikasi *Point of Sales* untuk Perusahaan Retail : Modul Persediaan Barang Dagang (Studi Kasus : CV Raka Putra Utama, Bandung)

1st Laura Audilia Simbolon
Fakultas Ilmu Terapan
Universitas Telkom
Bandung, Indonesia

laurasimbolon@student.telkomuni-
versity.ac.id

2nd Asti Widayanti
Fakultas Ilmu Terapan
Universitas Telkom
Bandung, Indonesia

astiwidayanti@telkomuniversity.ac.id

3rd Kastaman
Fakultas Ilmu Terapan
Universitas Telkom
Bandung, Indonesia

kastaman@telkomuniversity.ac.id

Abstrak – Penelitian ini dilatarbelakangi sering terjadinya kesalahan dalam proses pencatatan persediaan barang dagang, ketidaksesuaian antara jumlah barang fisik yang dicatat dan yang sebenarnya, dan pengolahan laporan keuangan atas transaksi yang terjadi belum terkomputerisasi, serta belum sesuai dengan pedoman akuntansi. Berbagai toko ritel, termasuk CV Raka Putra Utama, yang didirikan pada tahun 2019 dan berlokasi di Bandung mengalami masalah ini. Perancangan dilakukan dengan menggunakan metode *Software Development Life Cycle* (SDLC), yaitu metode *waterfall* serta menggunakan *Undefined Modelling Language* (UML). Sistem aplikasi diimplementasi dengan menggunakan Bahasa pemrograman PHP, CSS, HTML, dan menggunakan *framework CodeIgniter*. Kemudian dilakukan pengujian manual dan pengujian *Black Box Testing*. Pengujian menunjukkan bahwa aplikasi ini dapat menghasilkan laporan keuangan yang sesuai dengan standard akuntansi berdasarkan pencatatan transaksi yang terjadi.

Kata Kunci : *point of sales*, persediaan barang dagang, perusahaan retail, *codeigniter*

Abstract - This research is motivated by the frequent occurrence of errors in the process of recording merchandise inventory, discrepancies between the recorded and actual amounts of physical goods, and processing of financial reports for transactions that occur have not been computerized, and have not been in accordance with accounting guidelines. Various retail stores, including CV Raka Putra Utama, which was founded in 2019 and located in Bandung, are experiencing this problem. The design was carried out using the *Software Development Life Cycle* (SDLC) method, namely the *waterfall* method and using the *Undefined Modeling Language* (UML). The application system is implemented using the programming language PHP, CSS, HTML, and uses the *CodeIgniter* framework. Then manual testing and *Black Box Testing* were carried out. Tests show that this application can produce financial reports in accordance with accounting standards based on the recording of transactions that occur.

Keywords : *point of sales*, merchandise inventory, retail company, *codeigniter*

I. PENDAHULUAN

Persediaan barang dagang adalah suatu produk yang dimiliki serta disimpan oleh perusahaan dengan tujuan untuk dijual kembali, sehingga perusahaan dapat memperoleh keuntungan dari penjualan tersebut. Namun, terdapat beberapa masalah yang sering terjadi, seperti kerusakan barang, kesulitan menemukan tempat penyimpanan, selisih antara barang yang tersedia dengan yang tercatat, dan pencatatan yang belum terkomputerisasi rentan terjadi kesalahan serta belum sesuai dengan standard akuntansi. Perusahaan retail juga sering mengalami masalah ini.

CV Raka Putra Utama merupakan salah satu perusahaan di bidang retail pakaian konveksi yang beralamat di Jalan Kopo Katapang, Bandung, Jawa Barat. Perusahaan ini masih mencatat semua transaksi secara manual, termasuk pencatatan persediaan barang dagang menggunakan *excel*. Pencatatan yang masih manual atau belum terkomputerisasi menyebabkan pengelolaan persediaan barang dagang rentan terhadap kesalahan dalam pencatatan masuk dan keluarnya barang. Untuk mempermudah pengelolaan persediaan barang dagang, perusahaan dapat memanfaatkan fitur *scan barcode*. Dengan memanfaatkan fitur ini, diharapkan dapat meningkatkan keakuratan, kecepatan, dan efisiensi.

TABEL 1
Penelitian Terdahulu

No.	Judul Penelitian Penulis	
1.	Judul	Aplikasi <i>Point of Sales</i> Pada Perusahaan Retail (Studi Kasus : Perusahaan Artha Jaya, Banjarnegara)
	Penulis	Fanny Gunawan
	Tahun	2012
	Persamaan Proyek Akhir	Membuat Aplikasi <i>Point of Sales</i> (POS) pada toko retail yang terintegrasi dengan data barang pada sistem per gudangan dengan menggunakan <i>barcode scan</i> .

	Perbedaan Proyek Akhir	Aplikasi ini tidak menyediakan kartu stok.
2.	Judul	Aplikasi Berbasis Web untu <i>Point of Sales</i> (Studi Kasus : Unit Pengelolaan Kantin Universitas Telkom, Bandung)
	Penulis	Silvia Nurul Fithriyah
	Tahun	2018
	Persamaan Proyek Akhir	Membuat Aplikasi <i>Point of Sales</i> yang dapat mengelola persediaan barang dagang serta menghasilkan laporan dari transaksi yang terjadi.
	Perbedaan Proyek Akhir	Aplikasi ini tidak menyediakan fitur <i>barcode</i> .
3.	Judul	Aplikasi Pembelian, Penjualan, dan Persediaan Untuk Perusahaan Dagang (Studi Kasus di CV. Aman & Maju, Bandung)
	Penulis	Sahlida Febrianty
	Tahun	2018
	Persamaan Proyek Akhir	Membuat Aplikasi yang dapat mengelola Persediaan barang dagang berdasarkan transaksi yang terjadi.
	Perbedaan Proyek Akhir	Aplikasi ini menangani transaksi pembelian, pengelolaan pencatatan beban operasional, setoran modal awal, dan tidak memiliki fitur <i>barcode</i> .
4.	Judul	Sistem Informasi untuk Pembelian, Persediaan, dan Penjualan Berbasis Web (Studi Kasus : <i>Coffee Shope</i> Land Rocker, Bandung).
	Penulis	Anti Nur Fathimah
	Tahun	2022
	Persamaan Proyek Akhir	Membuat Aplikasi Pengelolaan Persediaan Barang Dagang
	Perbedaan Proyek Akhir	Aplikasi ini bukan merupakan Aplikasi <i>Point of Sales</i> yang terintegrasi dengan data barang dengan menggunakan <i>barcode scan</i> .
6.	Judul	Aplikasi Berbasis Web <i>Point of Sales</i> Warseda Menggunakan <i>Barcode</i> dengan Metode Perpetual (Studi Kasus : Koperasi Peternak Sapi Bandung Utara, Lembang).
	Penulis	Siti Hajjar
	Tahun	2022
	Persamaan Proyek Akhir	Membuat Aplikasi <i>Point of Sales</i> (POS) yang dapat mengelola persediaan barang dagang dengan menggunakan metode FIFO.
	Perbedaan Proyek Akhir	Aplikasi ini menyajikan daftar pengeluaran beban operasional.

II. METODE Pengerjaan

Berikut metode pengerjaan dalam pengumpulan data dan pengembangan aplikasi :

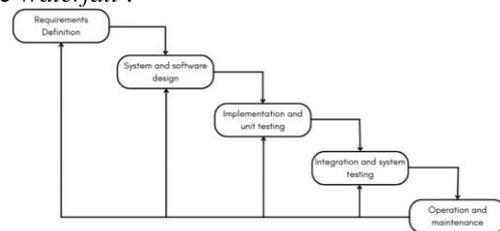
A. Metode Pengumpulan Data

Dalam pengumpulan data untuk proyek akhir ini dilakukan dengan cara, yaitu :

1. Wawancara
Merupakan metode pengumpulan informasi dengan mengajukan pertanyaan lisan kepada CV Raka Putra Utama. Proses ini dilakukan dengan tujuan mendapatkan informasi terkait pembangunan aplikasi.
2. Studi Kepustakaan
Merupakan metode pengumpulan informasi yang dilakukan dengan cara mengumpulkan dokumen yang berkaitan dengan masalah pengelolaan persediaan barang dan informasi mengenai aplikasi *Point of Sales*.

B. Metode Pengembangan Aplikasi

Dalam pengembangan Aplikasi menggunakan metode *Software Life Cycle* (SDLC), yaitu metode *waterfall*. SDLC adalah siklus yang digunakan untuk membangun suatu sistem informasi agar dapat berjalan dengan baik. Metode *waterfall* adalah pendekatan SDLC paling awal yang digunakan untuk pengembangan perangkat lunak. Hasil dari satu fase bertindak sebagai input fase berikutnya secara berurutan [1]. Berikut adalah contoh gambar SDLC dengan menggunakan metode *Waterfall* :



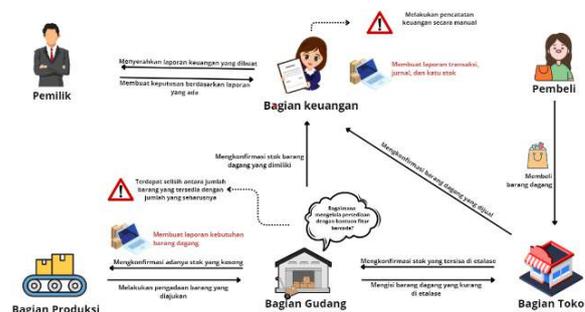
GAMBAR 1 Waterfall Diagram

III. ANALISIS & PERANCANGAN

Perancangan merupakan tahap awal dalam membangun sebuah aplikasi, pada tahap perancangan terdapat perancangan basis data dan perancangan aplikasi, Berikut merupakan perancangan dari basis data dan aplikasi tersebut.

A. Rich Picture

Berikut merupakan proses bisnis yang berjalan pada CV Raka Putra Utama.



GAMBAR 2 Rich Picture

Terdapat proses bisnis persediaan barang dagang pada CV Raka Putra Utama, berikut penjelasan singkat mengenai proses bisnis di atas.

1. Dimulai dari penjualan barang dagang yang dilakukan oleh bagian toko kepada pembeli. Setelah melakukan penjualan, bagian toko akan mengkonfirmasi barang yang terjual untuk dicatat oleh bagian keuangan. Bagian toko juga akan mengkonfirmasi stok yang tersisa di etalase jika barang yang tersedia tinggal sedikit atau sudah habis ke bagian gudang.
2. Bagian gudang akan menerima konfirmasi stok yang tersisa di etalase dari bagian toko. Setelah menerima konfirmasi tersebut, bagian gudang akan mengecek terlebih dahulu barang yang diminta, stok barang yang tersisa akan segera dikonfirmasi kepada bagian keuangan untuk dibuat laporan persediaan barang dagang. Jika barang sudah tidak tersedia di gudang, maka bagian gudang akan segera membuat laporan kebutuhan barang dagang. Laporan ini dibuat untuk mengkonfirmasi kepada bagian produksi bahwa terdapat stok yang hampir habis atau sudah habis.
3. Setelah menerima laporan kebutuhan barang dagang dari bagian gudang, maka bagian produksi akan segera memproduksi barang dagang yang dibutuhkan, barang yang telah diproduksi akan segera dikirimkan ke bagian gudang.
4. Saat menerima barang dagang, maka akan terbit laporan penerimaan barang dagang. laporan ini dibuat sebagai bukti bahwa barang yang diminta telah diterima oleh bagian gudang.
5. Dari transaksi yang terjadi, bagian keuangan akan membuat laporan mengenai transaksi-transaksi tersebut. Laporan yang telah dibuat akan diserahkan kepada pemilik untuk memberikan informasi mengenai transaksi yang terjadi serta menjadi salah satu pertimbangan pemilik dalam mengambil suatu keputusan.

B. Use Case Diagram

Berikut ini merupakan permodelan sistem dalam bentuk use case diagram yang terdapat pada aplikasi yang dibuat.



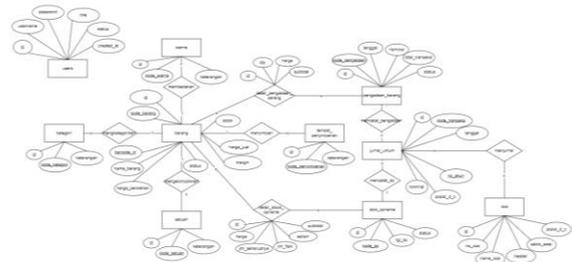
GAMBAR 3

Use Case Diagram

Terdapat 28 use case dengan tiga aktor, yaitu admin, bagian gudang, dan pemilik.

C. Entity Relationship Diagram (ERD)

Berikut analisis kebutuhan berbentuk Entity Relationship Diagram.



GAMBAR 4

Entity Relationship Diagram

IV. IMPLEMENTASI & PENGUJIAN

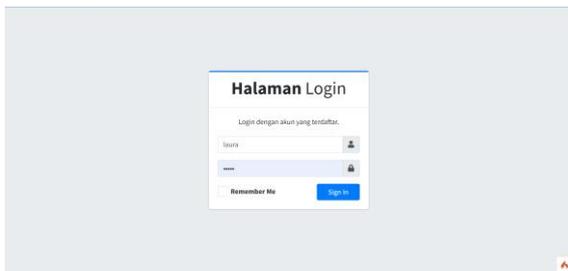
Tabir	Tindakan	Jenis	Penyortiran	Ukuran	Beban
barang	⌘ ⭐ 🗑️ 📄 📁 ➕ 🔍 🔄 🗑️	Koongan	Hapus	3 InnoDB utf8mb4_general_ci	48.9 KB
coa	⌘ ⭐ 🗑️ 📄 📁 ➕ 🔍 🔄 🗑️	Koongan	Hapus	9 InnoDB utf8mb4_general_ci	16.9 KB
detail_pengadaan_barang	⌘ ⭐ 🗑️ 📄 📁 ➕ 🔍 🔄 🗑️	Koongan	Hapus	22 InnoDB utf8mb4_general_ci	16.9 KB
detail_stok_opname	⌘ ⭐ 🗑️ 📄 📁 ➕ 🔍 🔄 🗑️	Koongan	Hapus	8 InnoDB utf8mb4_general_ci	16.9 KB
jurnal_umum	⌘ ⭐ 🗑️ 📄 📁 ➕ 🔍 🔄 🗑️	Koongan	Hapus	38 InnoDB utf8mb4_general_ci	16.9 KB
kartu_stok	⌘ ⭐ 🗑️ 📄 📁 ➕ 🔍 🔄 🗑️	Koongan	Hapus	49 InnoDB utf8mb4_general_ci	16.9 KB
kategori	⌘ ⭐ 🗑️ 📄 📁 ➕ 🔍 🔄 🗑️	Koongan	Hapus	4 InnoDB utf8mb4_general_ci	16.9 KB
log_transaksi	⌘ ⭐ 🗑️ 📄 📁 ➕ 🔍 🔄 🗑️	Koongan	Hapus	18 InnoDB utf8mb4_general_ci	16.9 KB
pengadaan_barang	⌘ ⭐ 🗑️ 📄 📁 ➕ 🔍 🔄 🗑️	Koongan	Hapus	17 InnoDB utf8mb4_general_ci	16.9 KB
satuan	⌘ ⭐ 🗑️ 📄 📁 ➕ 🔍 🔄 🗑️	Koongan	Hapus	3 InnoDB utf8mb4_general_ci	16.9 KB
stok_opname	⌘ ⭐ 🗑️ 📄 📁 ➕ 🔍 🔄 🗑️	Koongan	Hapus	9 InnoDB utf8mb4_general_ci	16.9 KB
tempat_penyimpanan	⌘ ⭐ 🗑️ 📄 📁 ➕ 🔍 🔄 🗑️	Koongan	Hapus	3 InnoDB utf8mb4_general_ci	16.9 KB
users	⌘ ⭐ 🗑️ 📄 📁 ➕ 🔍 🔄 🗑️	Koongan	Hapus	2 InnoDB utf8mb4_general_ci	16.9 KB
warna	⌘ ⭐ 🗑️ 📄 📁 ➕ 🔍 🔄 🗑️	Koongan	Hapus	3 InnoDB utf8mb4_general_ci	16.9 KB

GAMBAR 5

Implementasi Basis Data

A. Implementasi Proses Login

Berikut ini merupakan implementasi proses login yang berfungsi untuk masuk ke halaman utama aplikasi.



GAMBAR 6 Implementasi Login

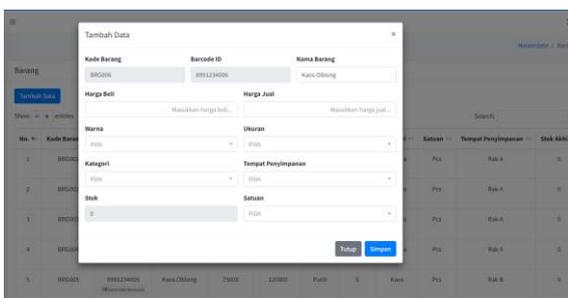
B. Implementasi Master Data Barang

Berikut ini merupakan implementasi proses menampilkan, menambah, dan mengubah master data barang. Pada tampilan master data barang memiliki tampilan daftar barang yang telah tersimpan di *database*, berfungsi untuk memberikan informasi mengenai data barang.

No. #	Kode Barang	Barcode ID	Nama Barang	Harga Beli	Harga Jual	Warna	Ukuran	Kategori	Satuan	Tempat Penyimpanan	Stok Awal
1	BRG001	8911234001	Celana Hitam	120000	180000	Hitam	S	Celana	Pcs	Rak A	0
2	BRG002	8911234002	Celana Hitam	120000	180000	Hitam	M	Celana	Pcs	Rak A	0
3	BRG003	8911234003	Celana Hitam	120000	180000	Hitam	L	Celana	Pcs	Rak A	0
4	BRG004	8911234004	Celana Hitam	120000	180000	Hitam	XL	Celana	Pcs	Rak A	0
5	BRG005	8911234005	Kaos Oblong	75000	120000	Putih	S	Kaos	Pcs	Rak B	0

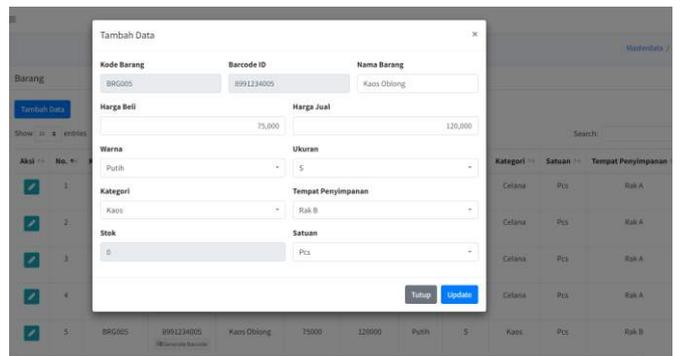
GAMBAR 7 Implementasi Daftar Master Data Barang

Untuk menambah data barang, admin perlu menekan tombol “Tambah Data” disebelah kiri atas. Setelah itu form data akan ditampilkan. Pada form tersebut admin dapat menginput data sesuai dengan kebutuhan. Pada bagian ini kode barang dan *barcode* sudah terinput secara otomatis sehingga tidak perlu melakukan penginputan untuk bagian ini. Ketika data sudah diinputkan, admin perlu menekan tombol simpan.



GAMBAR 8 Implementasi Tambah Master Data Barang

Ketika data barang sudah tersimpan namun, terdapat informasi yang akan diubah maka admin dapat mengubah data dengan cara menekan tombol “Edit”. Pada form edit terdapat beberapa data yang tidak dapat diubah, yaitu kode barang, *barcode* id, dan stok. Ketika data sudah sesuai maka admin dapat menekan tombol *update*.



GAMBAR 9 Implementasi Ubah Master Data Barang

C. Implementasi Master Data COA

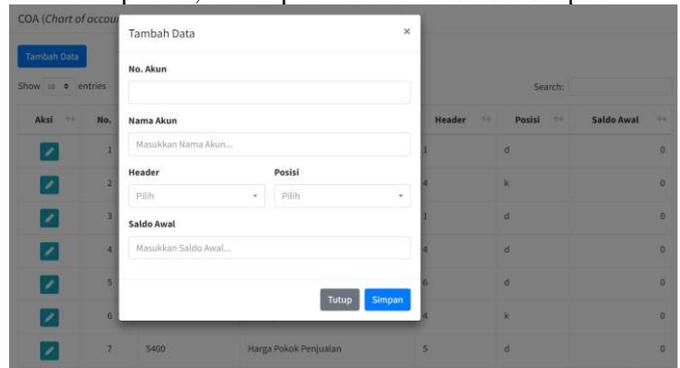
Berikut ini merupakan implementasi proses menampilkan, menambah, dan mengubah master data COA. Pada tampilan master data COA memiliki tampilan daftar COA yang telah tersimpan di *database*, berfungsi untuk memberikan informasi mengenai data COA.

COA (Chart of account)

Aksi	No. #	Kode Akun	Nama Akun	Header	Posisi	Saldo Awal
<input checked="" type="checkbox"/>	1	1110	Kas	1	d	0
<input checked="" type="checkbox"/>	2	4110	Penjualan	4	k	0
<input checked="" type="checkbox"/>	3	1130	Persediaan Barang Dagang	1	d	0
<input checked="" type="checkbox"/>	4	4140	Persediaan lain-lain	4	d	0
<input checked="" type="checkbox"/>	5	6110	Biaya Lain-lain	6	d	0
<input checked="" type="checkbox"/>	6	4130	Potongan Penjualan	4	k	0
<input checked="" type="checkbox"/>	7	5400	Harga Pokok Penjualan	5	d	0

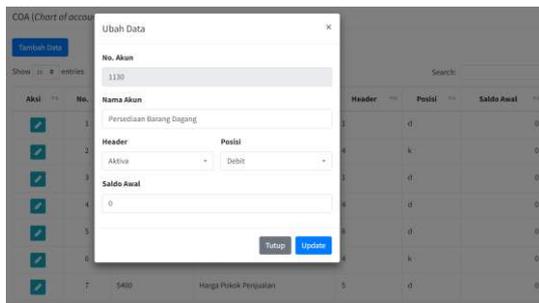
GAMBAR 10 Implementasi Daftar Master Data COA

Untuk menambah data COA, admin perlu menekan tombol “Tambah Data” disebelah kiri atas. Setelah itu form tambah data akan ditampilkan. Pada form tersebut admin dapat menginput data sesuai dengan kebutuhan. Ketika data sudah diinputkan, admin perlu menekan tombol simpan.

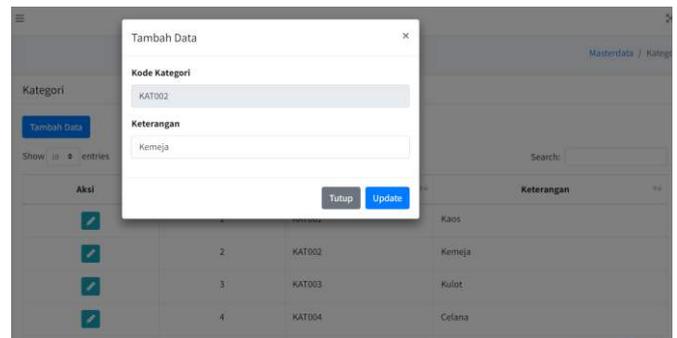


GAMBAR 11 Implementasi Tambah Master Data COA

Ketika data COA sudah tersimpan namun, terdapat informasi yang akan diubah maka admin dapat mengubah data dengan cara menekan tombol “Edit”. Pada form edit terdapat data yang tidak dapat diubah, yaitu nomor akun. Ketika data sudah sesuai maka admin dapat menekan tombol *update*.



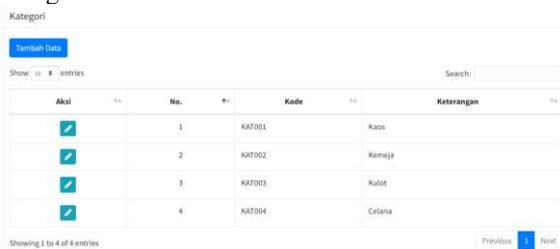
GAMBAR 12
Implementasi Ubah Master Data COA



GAMBAR 15
Implementasi Ubah Master Data Kategori

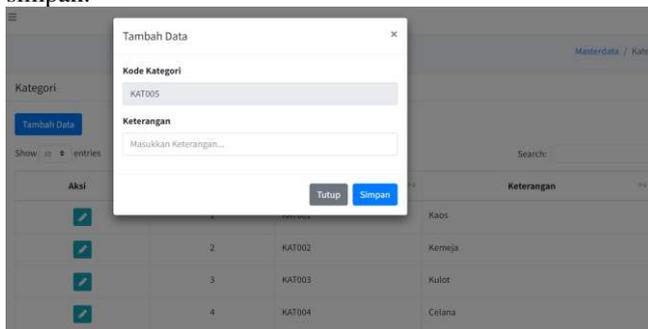
D. Implementasi Master Data Kategori

Berikut ini merupakan implementasi proses menampilkan, menambah, dan mengubah master data kategori. Pada tampilan master data kategori memiliki tampilan daftar kategori yang telah tersimpan di *database*, berfungsi untuk memberikan informasi mengenai data kategori.



GAMBAR 13
Implementasi Daftar Master Data Kategori

Untuk menambah data Kategori, admin perlu menekan tombol “Tambah Data” disebelah kiri atas. Setelah itu form tambah data akan ditampilkan. Pada form tersebut admin dapat menginput data sesuai dengan kebutuhan. Ketika data sudah diinputkan, admin gudang perlu menekan tombol simpan.



GAMBAR 14
Implementasi Tambah Master Data Kategori

Ketika data kategori sudah tersimpan namun, terdapat informasi yang akan diubah maka admin dapat mengubah data dengan cara menekan tombol “Edit”. Pada form edit terdapat data yang tidak dapat diubah, yaitu kode kategori. Ketika data sudah sesuai maka admin dapat menekan tombol *update*.

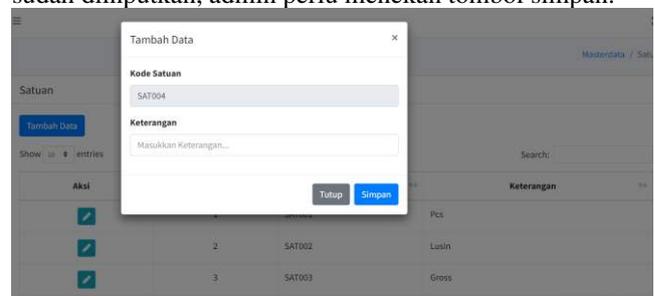
E. Implementasi Master Data Satuan

Berikut ini merupakan implementasi proses menampilkan, menambah, dan mengubah master data satuan. Pada tampilan master data satuan memiliki tampilan daftar satuan yang telah tersimpan di *database*, berfungsi untuk memberikan informasi mengenai data satuan.



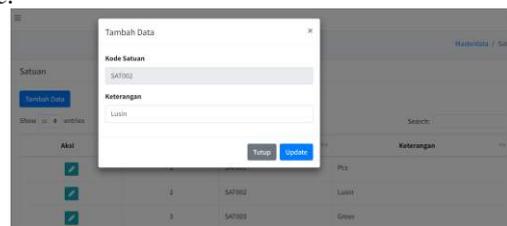
GAMBAR 16
Implementasi Daftar Master Data Satuan

Untuk menambah data Satuan, admin perlu menekan tombol “Tambah Data” disebelah kiri atas. Setelah itu form tambah data akan ditampilkan. Pada form tersebut admin dapat menginput data sesuai dengan kebutuhan. Ketika data sudah diinputkan, admin perlu menekan tombol simpan.



GAMBAR 17
Implementasi Tambah Master Data Satuan

Ketika data satuan sudah tersimpan namun, terdapat informasi yang akan diubah maka admin dapat mengubah data dengan cara menekan tombol “Edit”. Pada form edit terdapat data yang tidak dapat diubah, yaitu kode satuan. Ketika data sudah sesuai maka admin dapat menekan tombol *update*.



GAMBAR 18
Implementasi Ubah Master Data Satuan

F. Implementasi Master Data Tempat Penyimpanan

Berikut ini merupakan implementasi proses menampilkan, menambah, dan mengubah master data tempat penyimpanan. Pada tampilan master data tempat penyimpanan memiliki tampilan tempat penyimpanan yang telah tersimpan di *database*, berfungsi untuk memberikan informasi mengenai data tempat penyimpanan.

Aksi	No.	Kode	Keterangan
<input checked="" type="checkbox"/>	1	PNV001	Rak A
<input checked="" type="checkbox"/>	2	PNV002	Rak B
<input checked="" type="checkbox"/>	3	PNV003	Rak C
<input checked="" type="checkbox"/>	4	PNV004	Rak D

GAMBAR 19

Implementasi Daftar Master Data Tempat Penyimpanan

Untuk menambah data tempat penyimpanan, admin perlu menekan tombol “Tambah Data” disebelah kiri atas. Setelah itu form tambah data akan ditampilkan. Pada form tersebut admin dapat menginput data sesuai dengan kebutuhan. Ketika data sudah diinputkan, admin perlu menekan tombol simpan.

GAMBAR 20

Implementasi Tambah Master Data Tempat Penyimpanan

Ketika data tempat penyimpanan sudah tersimpan namun, terdapat informasi yang akan diubah maka admin dapat mengubah data dengan cara menekan tombol “Edit”. Pada form edit terdapat data yang tidak dapat diubah, yaitu kode penyimpanan. Ketika data sudah sesuai maka admin dapat menekan tombol *update*.

GAMBAR 21

Implementasi Ubah Master Data Tempat Penyimpanan

G. Implementasi Master Data Warna

Berikut ini merupakan implementasi proses menampilkan, menambah, dan mengubah master data warna. Pada tampilan master data warna memiliki tampilan daftar warna yang telah tersimpan di *database*, berfungsi untuk memberikan informasi mengenai warna seperti berikut.

Aksi	No.	Kode	Keterangan
<input checked="" type="checkbox"/>	1	WRN001	Hitam
<input checked="" type="checkbox"/>	2	WRN002	Navy
<input checked="" type="checkbox"/>	3	WRN003	Abu-Abu
<input checked="" type="checkbox"/>	4	WRN004	Putih
<input checked="" type="checkbox"/>	5	WRN005	Cream

GAMBAR 22

Implementasi Daftar Master Data Warna

Untuk menambah data warna, admin perlu menekan tombol “Tambah Data” disebelah kiri atas. Setelah itu form tambah data akan ditampilkan. Pada form tersebut admin dapat menginput data sesuai dengan kebutuhan. Ketika data sudah diinputkan, admin perlu menekan tombol simpan.

GAMBAR 23

Implementasi Tambah Master Data Kategori

Ketika data warna sudah tersimpan namun, terdapat informasi yang akan diubah maka admin dapat mengubah data dengan cara menekan tombol “Edit”. Pada form edit terdapat data yang tidak dapat diubah, yaitu kode warna. Ketika data sudah sesuai maka admin dapat menekan tombol *update*.

GAMBAR 24

Implementasi Ubah Master Data Warna

H. Implementasi Transaksi Pengadaan Barang

Berikut ini merupakan implementasi proses menampilkan dan menambah transaksi pengadaan barang. Pada tampilan transaksi pengadaan barang memiliki tampilan pengadaan barang yang telah tersimpan di *database*, berfungsi untuk memberikan informasi mengenai data pengadaan barang.

No	Kode Pengadaan	Tgl Pengadaan	Nominal	Total Transaksi	Kembalian	Status
1	PBRG20230130040	2023-01-30	15.000	15.000	0	1
2	PBRG20230128009	2023-01-28	100.000	75.000	25.000	1
3	PBRG20230128008	2023-01-28	45	45	0	1
4	PBRG20230128007	2023-01-28	100	100	0	1
5	PBRG20230128006	2023-01-28	25	25	0	1
6	PBRG20230128005	2023-01-28	105	105	0	1
7	PBRG20230130004	2023-01-30	5.970.000	5.970.000	0	1
8	PBRG20230130003	2023-01-30	250.000	250.000	0	1

GAMBAR 25

Implementasi Daftar Transaksi Pengadaan Barang

Untuk menambah data transaksi pengadaan barang, admin gudang perlu menekan tombol “Tambah Data” disebelah kiri atas. Setelah itu form tambah data akan ditampilkan. Pada form tersebut admin gudang dapat menginput data sesuai dengan kebutuhan. Pada bagian ini id pengadaan, tanggal pengadaan, harga barang, dan jumlah sudah terinput secara otomatis sehingga tidak perlu melakukan penginputan untuk bagian ini. Ketika data sudah diinputkan, admin gudang perlu menekan tombol simpan.

GAMBAR 26

Implementasi Tambah Transaksi Pengadaan Barang

I. Implementasi *Stok Opname*

Berikut ini merupakan implementasi proses menampilkan dan menambah *stock opname*. Pada tampilan *stock opname* memiliki tampilan *stock opname* yang telah tersimpan di *database*, berfungsi untuk memberikan informasi mengenai *stock opname* seperti berikut.

No	ID Stok Opname	Tgl Stok Opname	Jumlah Seharusnya	Jumlah Fisik
1	SO20230712001	2023-07-12	70	70
2	SO20230712002	2023-07-12	25	22
3	SO20230712003	2023-07-12	2	5

GAMBAR 27

Implementasi Daftar Transaksi Stock Opname

Untuk menambah data *stock opname*, admin gudang perlu menekan tombol “Tambah Data” disebelah kiri atas. Setelah itu form tambah data akan ditampilkan. Pada form tersebut admin gudang dapat menginput data sesuai dengan kebutuhan. Pada bagian ini kode so dan tanggal so sudah terinput secara otomatis sehingga tidak perlu melakukan penginputan untuk bagian ini. Ketika data sudah diinputkan, admin gudang perlu menekan tombol simpan.

GAMBAR 28

Implementasi Tambah Transaksi Stok Opname

J. Implementasi Laporan Pengadaan Barang

Berikut ini merupakan implementasi laporan pengadaan barang dari transaksi pengadaan yang telah terjadi.

No	Kode Pengadaan	Tgl Pengadaan	Nominal	Total Transaksi	Kembalian
1	PBRG20230712002	2023-07-12	0	150.000	0
2	PBRG20230712001	2023-07-12	0	8.400.000	0

GAMBAR 29

Implementasi Laporan Pengadaan Barang

K. Implementasi Laporan *Stok Opname*

Berikut ini merupakan implementasi laporan *stock opname* dari transaksi pengadaan yang telah terjadi.

No	ID Stok Opname	Tgl Stok Opname	Jumlah Seharusnya	Jumlah Fisik
1	SO20230712001	2023-07-12	70	70
2	SO20230712002	2023-07-12	25	22
3	SO20230712003	2023-07-12	2	5

GAMBAR 30

Implementasi Laporan Stok Opname

L. Implementasi Jurnal Umum

Berikut ini merupakan implementasi jurnal umum dari transaksi yang telah terjadi.

No	Kode Transaksi	Tanggal	No Akun	Keterangan	Debet	Kredit
1	PBRG20230600001	2023-06-01	1130	Persediaan Barang Dagang	2.700.000	
2		2023-06-01	1110	Kas		2.700.000
3	SO20230801011	2023-08-01	6110	Biaya Lain-lain	75.000	
4		2023-08-01	1130	Persediaan Barang Dagang		75.000
5	SO20230801012	2023-08-01	6110	Biaya Lain-lain	120.000	
6		2023-08-01	1130	Persediaan Barang Dagang		120.000
TOTAL					2.895.000	2.895.000

GAMBAR 31

Implementasi Jurnal Umum

M. Implementasi Buku Besar

Berikut ini merupakan implementasi buku besar dari jurnal umum.

Tanggal	Nama Akun	Ref	Nominal		
			Debet	Kredit	Saldo
0000-00-00	Saldo Awal				Rp 0
2023-07-12	Persediaan Barang Dagang	1130	Rp 8,400,000		Rp 8,400,000
2023-07-12	Persediaan Barang Dagang	1130		Rp 960,000	Rp 8,040,000

GAMBAR 32
Implementasi Buku Besar

N. Implementasi Kartu Stok

Berikut ini merupakan implementasi kartu stok dari transaksi yang telah terjadi.

Tanggal	ID Transaksi	Stok Masuk			Stok Keluar			Stok Total		
		Unit	Harga/unit	Total	Unit	Harga/unit	Total	Unit	Harga/unit	Total
2023-07-12	PBRG20230712001	25	120,000	3,000,000	0	0	0	25	120,000	3,000,000

GAMBAR 33
Implementasi Kartu Stok

V. KESIMPULAN

Berdasarkan pembahasan yang telah diuraikan diatas, maka dapat diambil kesimpulan bahwa:

- Aplikasi pada proyek akhir ini dapat melakukan pendataan persediaan barang dagang,
- Aplikasi pada proyek akhir ini dapat mengeneralisasi *barcode* barang,
- Aplikasi pada proyek akhir ini dapat mengelola data persediaan barang dagang,
- Aplikasi pada proyek akhir ini dapat mengelola stok persediaan barang dagang dari transaksi pengadaan,
- Aplikasi pada proyek akhir ini dapat mencatat selisih antara barang yang tercatat dengan barang yang tersedia,
- Aplikasi pada proyek akhir ini dapat menampilkan jurnal umum, buku besar, dan kartu stok.

REFERENSI

- [1] Indra Rianto, *Perancangan dan Pengembangan Penentuan Uang Kuliah Tunggal Menggunakan CodeIgniter*. Lakeisha, 2021.